LHCb 2001-021 TRAC
March 13, 2001

Addition of the vertex detector measurements to the track fit

Rutger van der Eijk, Rutger Hierck, Marcel Merk,
Matthew Needham.

NIKHEF, Amsterdam, The Netherlands

Abstract

This note describes the inclusion of vertex detector measurements to the LHCb object oriented track fit.

Contents

1 Introduction

The note [1] describes the performance of the LHCb Object Oriented track fitting software. In the described fit no vertex detector information was used because the necessary code was not yet available in C++. This code has been implemented since September 2000.

Two new measurement types (i.e. classed derived from TrMeasurement), one corresponding to the vertex detector R clusters (class VeloRClusterOnTrack) and one to the vertex detector ϕ clusters (class VeloPhiClusterOnTrack), have been implemented. The track building algorithm (class TrTracksCreator) is extended with the possibility to include these vertex detector measurements to the track. The track fitting algorithms work with any TrMeasurement. Therefore adding the vertex detector measurement objects to the track automatically results in a track fit that includes the vertex detector.

This note describes the performance of the track fit using these vertex detector measurements. Recently a new VELO geometry has been proposed [2]. The track fit software has to be updated to use the measurements of this geometry.

2 Vertex Detector Measurements

There are two different types of vertex detectors, the vertex R- and vertex ϕ- detector. The simulation and reconstruction of the vertex detector in SICB is described in [3]. Within the Gaudi framework the output of the reconstruction, that is to say vertex R- and ϕ-clusters (both represented by the class VeloCluster) are available from the event store as described in [4]. This section describes how these clusters are used in the track fit.

2.1 R clusters

Vertex detector R-clusters measure the distance to the origin r (see figure 1) at which a particle traversed the detector plane. Hence the measurement m is defined as

$$m = r$$

where r is the radial distance of the reconstructed cluster in the vertex detector.

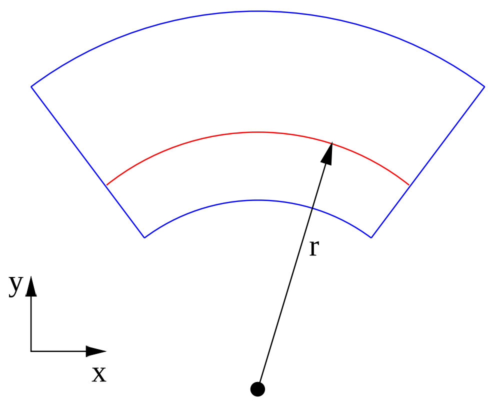

Figure 1: Projection of track parameters into the measurement space of a vertex detector R-cluster.

In order to use the R clusters in the track fit we need to define how a track state $\mathbf{x} = (\mathrm{x}, \mathrm{y}, \mathrm{t_x}, \mathrm{t_y}, \mathrm{Q/P})$ is projected into the measurement space of the detector[1]. The projection relation in terms of the track state $\mathbf{x}$ is

$$m_{pred} = h(\mathbf{x}) = \sqrt{\mathrm{x}^2 + \mathrm{y}^2}$$

Therefore the track projection matrix elements are

$$\partial h / \partial x = \frac{x}{\sqrt{x^2 + y^2}}$$

$$\partial h / \partial y = \frac{y}{\sqrt{x^2 + y^2}}$$

$$\partial h / \partial t_x = 0$$

$$\partial h / \partial t_y = 0$$

$$\partial h / \partial \kappa = 0$$

Within the TRAIL software a new measurement type derived from the TrMeasurement class, corresponding to a vertex detector R-cluster is implemented in the class VeloRClusterOnTrack. In figure 2 the pull distribution of

[1]See [5] for an explanation on the Kalman filter, track states, projections etc. Furthermore we use the same notation as in this note.

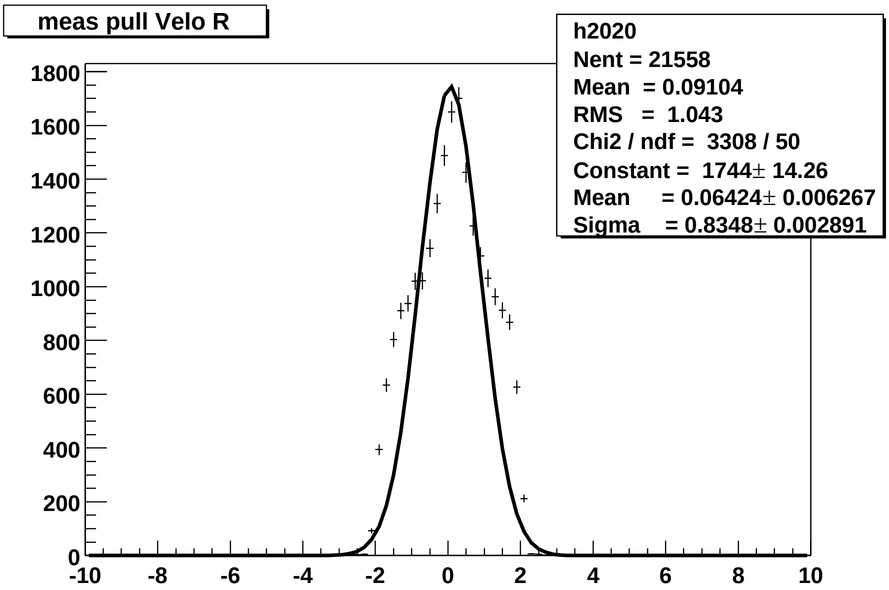

Figure 2: Pull distribution of the measurement value of vertex detector R-clusters, i.e. $(r - r_{true})/\sigma_r$.

the measurement value of these R-clusters, i.e. $(r - r_{true})/\sigma_r$, is shown. The distribution is clearly non Gaussian as can be expected. A single strip being hit gives a resolution of pitch/$\sqrt{12}$. If two or more strips are hit the center of gravity can be used to give an improved measurement position. Hence the observed shape. Therefore we can expect non Gaussian effects in the fitted track parameters if vertex detector R-clusters are taken into account.

2.2 ϕ clusters

Vertex detector ϕ-clusters measure the angle ϕ^t or "almost ϕ". The angle ϕ^t as shown in figure 3 is the ϕ coordinate the hit strip has at the inner radius of the silicon wafer. There is a difference between ϕ^t and ϕ because the strips are tilted under an angle α_t.

Instead of using the angle ϕ^t in the track fit we use a distance as the measured quantity. In the track fit a vertex ϕ-detector measurement is defined as the shortest distance d to the imaginary strip through the origin under the angle ϕ^t. From figure 3 it becomes clear that the measurement m then satisfies

$$m = d = R\sin(\alpha_t)$$

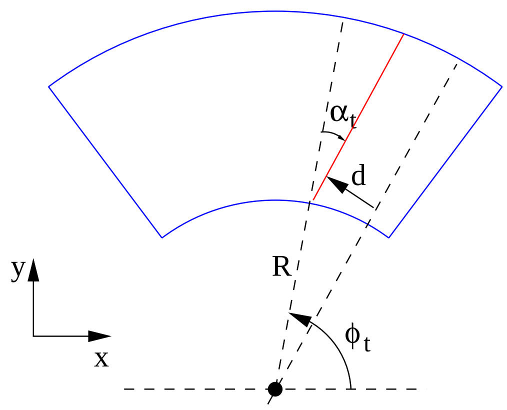

Figure 3: Projection of track parameters into the measurement space of a vertex detector ϕ cluster.

where R is the inner radius of the silicon wafer. At first sight this looks surprising because there is no dependency on the angle ϕ^t, i.e. the measurement value is the same for all clusters. However the measurement does determine which strip is hit and hence the angle ϕ^t. This angle substracted by α_t is the angle the strip makes with the x-axis. The projection of the track state on the strip directly uses this angle.

The projection relation in terms of a track state $\mathbf{x}$ is

$$m_{pred} = h(\mathbf{x}) = -\mathrm{x}\sin(\phi_\mathrm{t} - \alpha_\mathrm{t}) + \mathrm{y}\cos(\phi_\mathrm{t} - \alpha_\mathrm{t})$$

Therefore the track projection matrix elements are

$$\partial h/\partial x = -\sin(\phi_t - \alpha_t)$$

$$\partial h/\partial y = \cos(\phi_t - \alpha_t)$$

$$\partial h/\partial t_x = 0$$

$$\partial h/\partial t_y = 0$$

$$\partial h/\partial \kappa = 0$$

Within the TRAIL software a new measurement type corresponding to a vertex detector ϕ-cluster is implemented in the class VeloPhiClusterOnTrack. In figure 4 the pull distribution of the measurement value of these ϕ-clusters, i.e. $(d - d_{true})/\sigma_d$, is shown. Again this distribution is non Gaussian. Therefore we can expect non Gaussian effects in the fitted track parameters if vertex detector ϕ-clusters are taken into account.

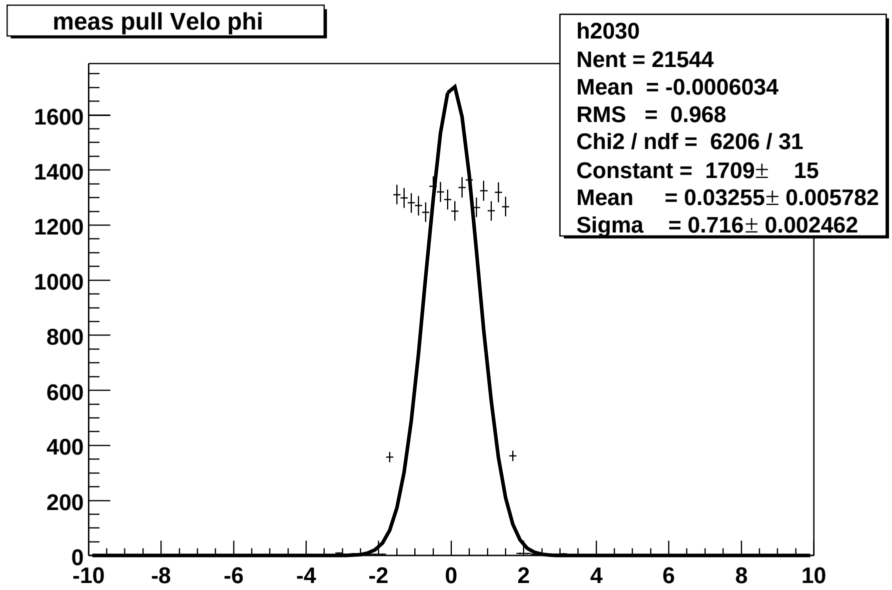

Figure 4: Pull distribution of the measurement value of vertex detector ϕ-clusters, i.e. $(d - d_{true})/\sigma_d$.

3 Fit Performance

This section describes the results of the track fit when both the vertex R-clusters as well as the ϕ-clusters are included in the fit. A sample of 150 generic $b\bar{b}$ events were generated with SICBMC[6] v233r2 with database v229r2, the latter corresponding to an aluminium beampipe. The data were reconstructed with Brunel v1r6[7] and the TRAIL tracking software. A measure of the of the reliability of the fit are the pull distributions, namely the difference of the reconstructed and corresponding Monte Carlo quantity divided by the calculated error. If all the errors are Gaussian and properly taken into account each pull should follow a normal distribution centred on zero with unit variance.

In the results presented in [1] we fitted tracks up to the first measurement position on the track, which usually corresponded to an Inner Tracker hit in station 1. Now we fit all the way to the vertex detector[2] and hence more material is encountered. It turned out that in order to obtain a momentum pull centrered on zero the dE/dx correction factor c_{ion}[1] needed to be retuned. A value of 60 MeV was found for c_{ion}.

[2]Note that we have performed an upstream fit.

Another surprising observation was a dependence of the position pulls in the vertex detector as a function of the z-position. It turned out this was caused by the small value of the magnetic field still present in the vertex detector. This was first assumed to have a negligible effect, but this turned out not to be the case. In order to obtain good pulls for the position coordinates it was necessary to use an extrapolation algorithm that took into account the local magnetic field.

		x	y	t_x	t_y
Outer Tracker	Pull	1.01	1.02	0.98	1.01
	Resolution	$56\mu m$	$134\mu m$	1.3×10^{-4}	1.9×10^{-4}
Inner Tracker	Pull	1.06	1.15	1.01	1.08
	Resolution	$45\mu m$	$95\mu m$	0.83×10^{-4}	1.0×10^{-4}
Vertex R	Pull	1.14	1.13	1.41	1.28
	Resolution	$9.1\mu m$	$8.4\mu m$	1.0×10^{-4}	1.0×10^{-4}
Vertex ϕ	Pull	1.14	1.14	1.43	1.36
	Resolution	$9.0\mu m$	$8.6\mu m$	1.0×10^{-4}	1.0×10^{-4}

Table 1: Core resolution and pull distribution of a fit to the track parameters distributions for the different detector measurements. Note that for the resolution distributions a double Gaussian does not always give a perfect fit.

We present the pull distributions of the track parameters x, y, t_x and t_y at the measurement positions for the four detector types. No momentum pulls are given at these z-positions because the "true" momentum was only known at the track vertex. Furthermore we present the resolution distribution of these parameters. In figure 5 the pull distribution of the track parameters x, y, t_x and t_y at the z-positions of each measurement in the Outer Tracker are found. Figure 6 shows the resolution of these four track parameters at the same z-positions in the Outer Tracker. Similar plots for the Inner Tracker can be found in figure 7 and figure 8. For the vertex R-detector see the figures 9 and 10. For the vertex ϕ-detector see the figures 11 and 12. In table 1 the results are summarised. The shown values are the σ of the core Gaussian of a double Gaussian fit through the distributions.

In figure 13 the pull distributions of the track parameters x, y, t_x and t_y at the true vertex position of the track, i.e. the z-position where the particle was created in the simulation are shown. Figure 14 shows the resolution distributions of these track parameters at the track true vertex z-position. At the vertex position the true track momentum is well known. In figure 15 you find the momentum pull distribution. Also shown is the momentum

	x	y	t_x	t_y	p^3
Pull	1.13	1.10	0.94	0.83	1.16
Resolution	$28\mu m$	$26\mu m$	1.3×10^{-4}	1.4×10^{-4}	3.6×10^{-3}

Table 2: Core resolution and pull distribution of a fit to the track parameters distributions at the true track vertex position.

resolution distribution. The core momentum resolution is around 3.6×10^{-3}. This is consistent with the results presented in [1].

4 Conclusions

The following conclusions can be drawn:

- Two new measurement types corresponding to the vertex R- and vertex ϕ- clusters have been implemented in C++.
- This allows the vertex detector measurements to be taken into account in the TRAIL track fit. With this addition all the functionality of the SICB fit is provided in the C++ version.
- The core momentum resolution found is 3.6×10^{-3} which is consistent with resolution presented in [1].
- The single track position resolution at the track creation vertex is about $27\mu m$.
- The single track angle resolution at the track creation vertex is about 1.4×10^{-4}rad.

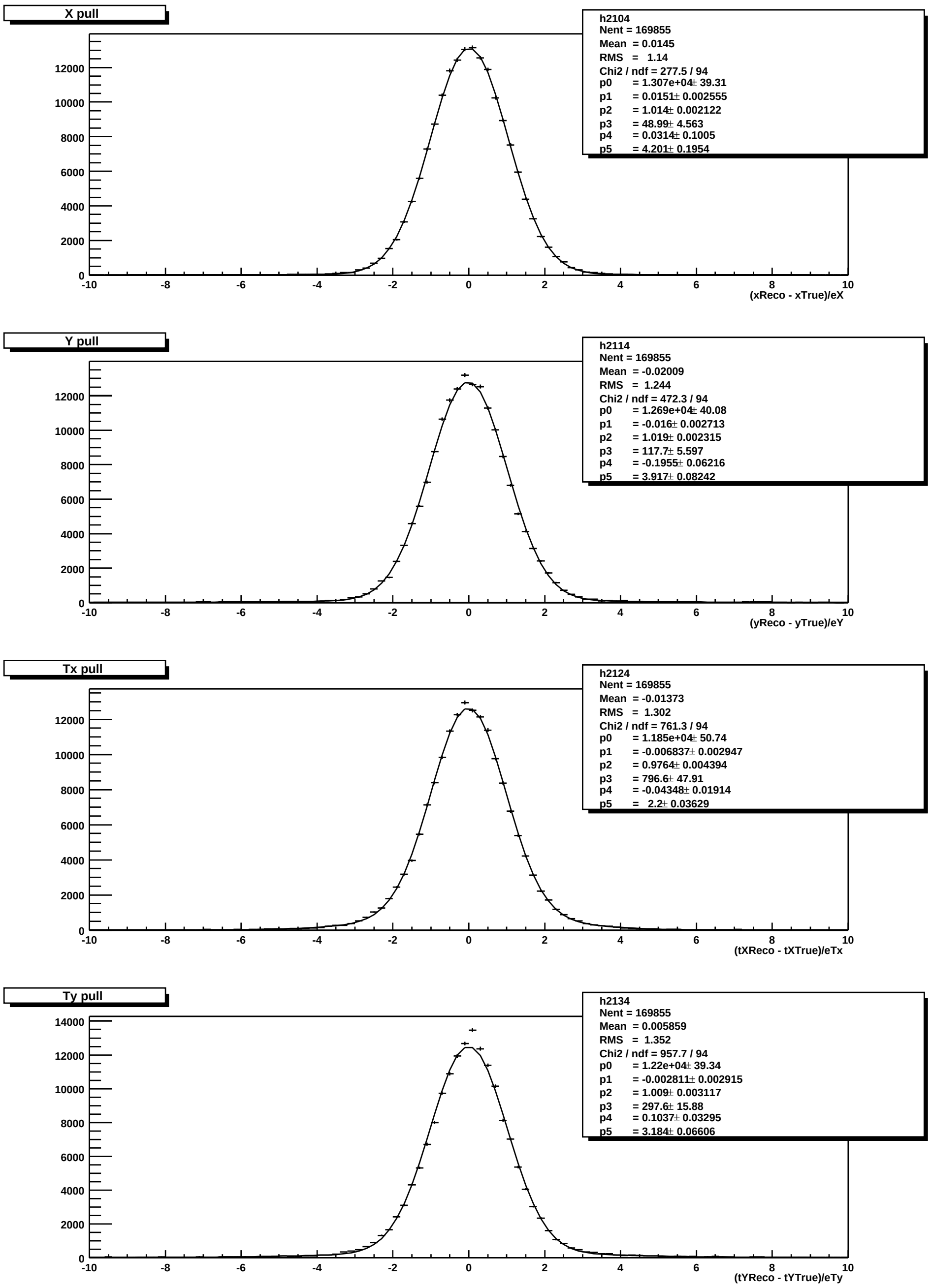

Figure 5: Pull distribution of the track parameters x, y, t_x, and t_y at the Outer Tracker hit positions. The fitted curve is a double Gaussian.

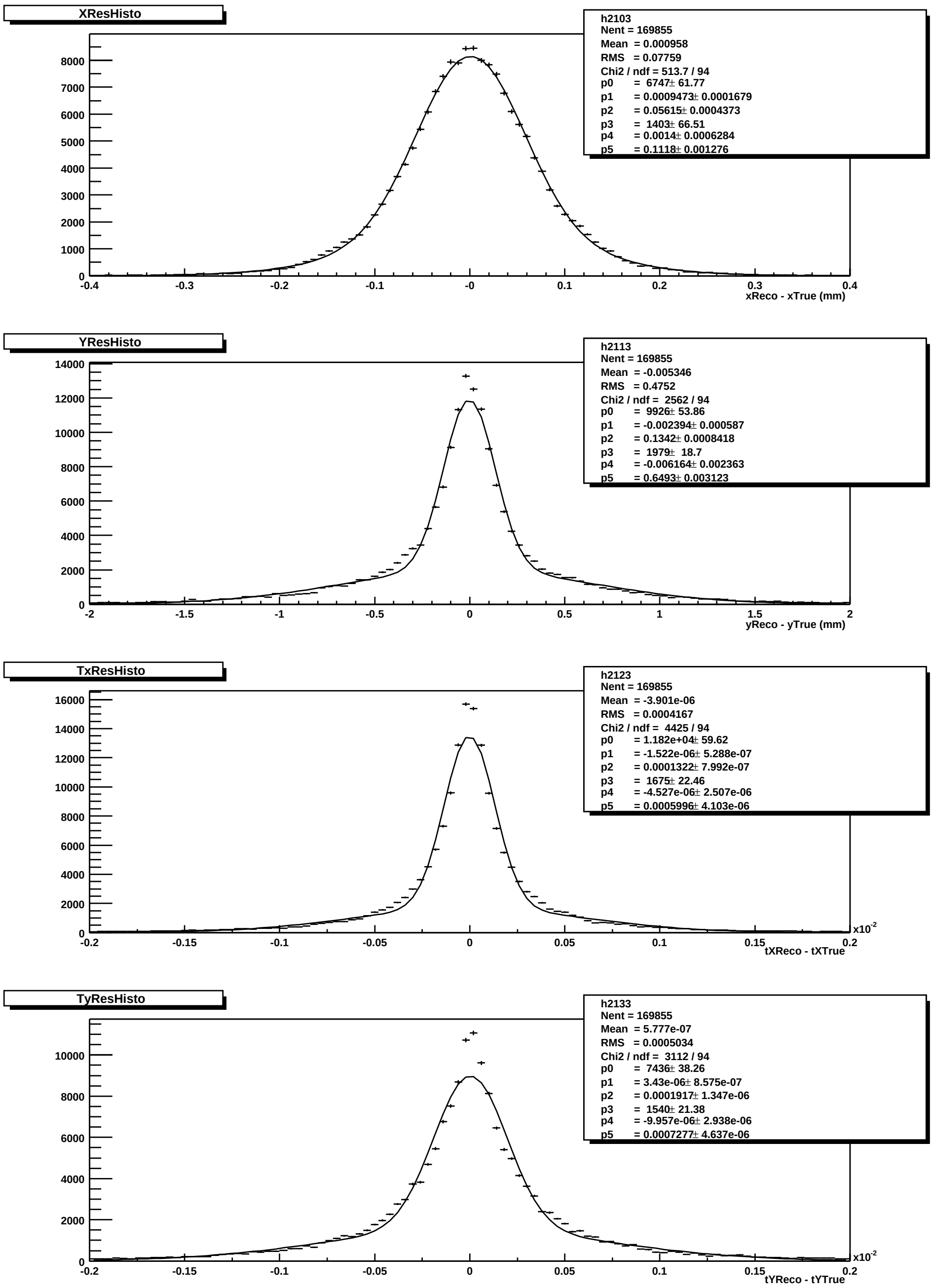

Figure 6: Resolution distribution of the track parameters x, y, t_x, and t_y at the Outer Tracker hit positions. The fitted curve is a double Gaussian.

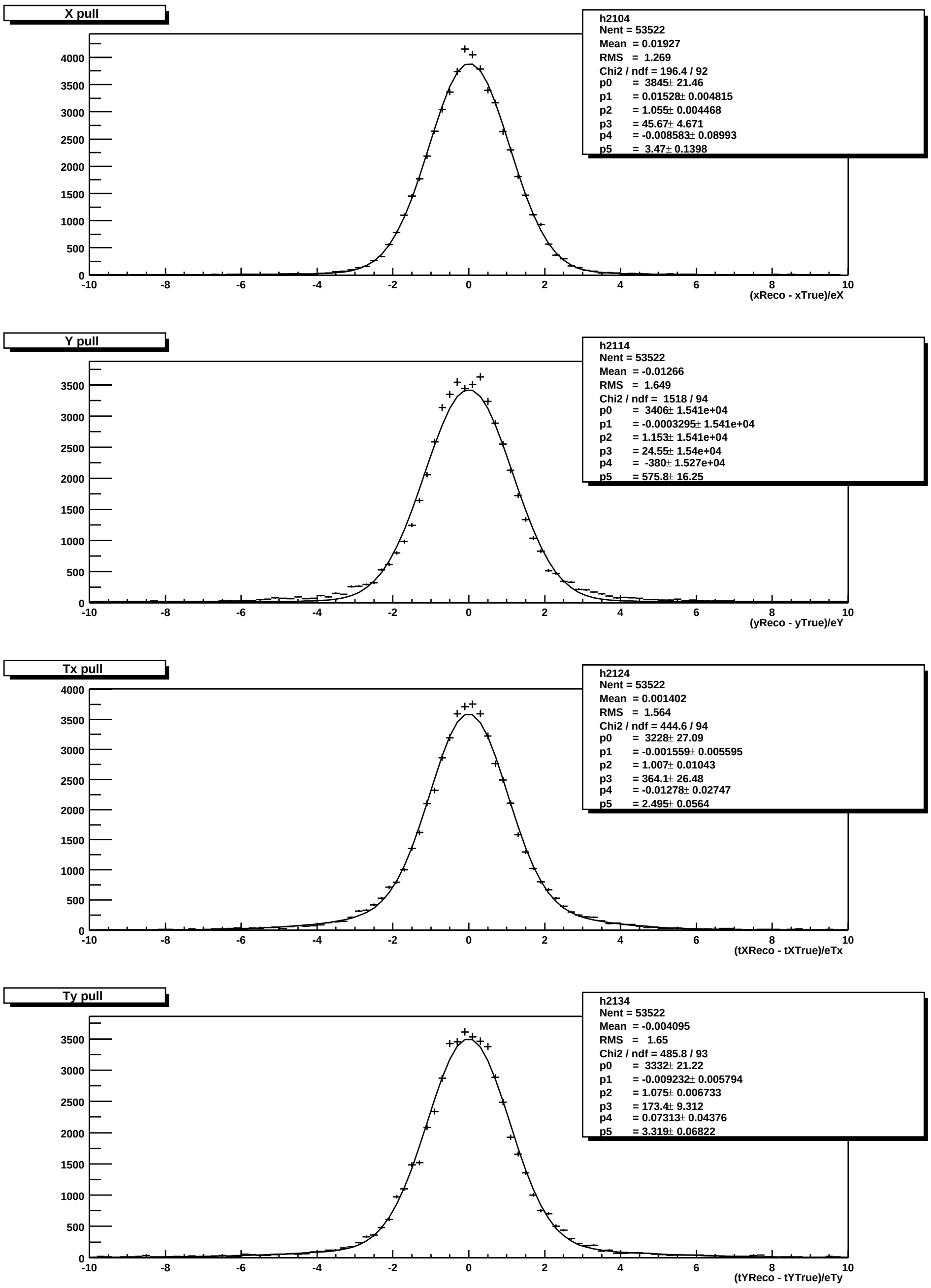

Figure 7: Pull distribution of the track parameters x, y, t_x, and t_y at the Inner Tracker hit positions. The fitted curve is a double Gaussian.

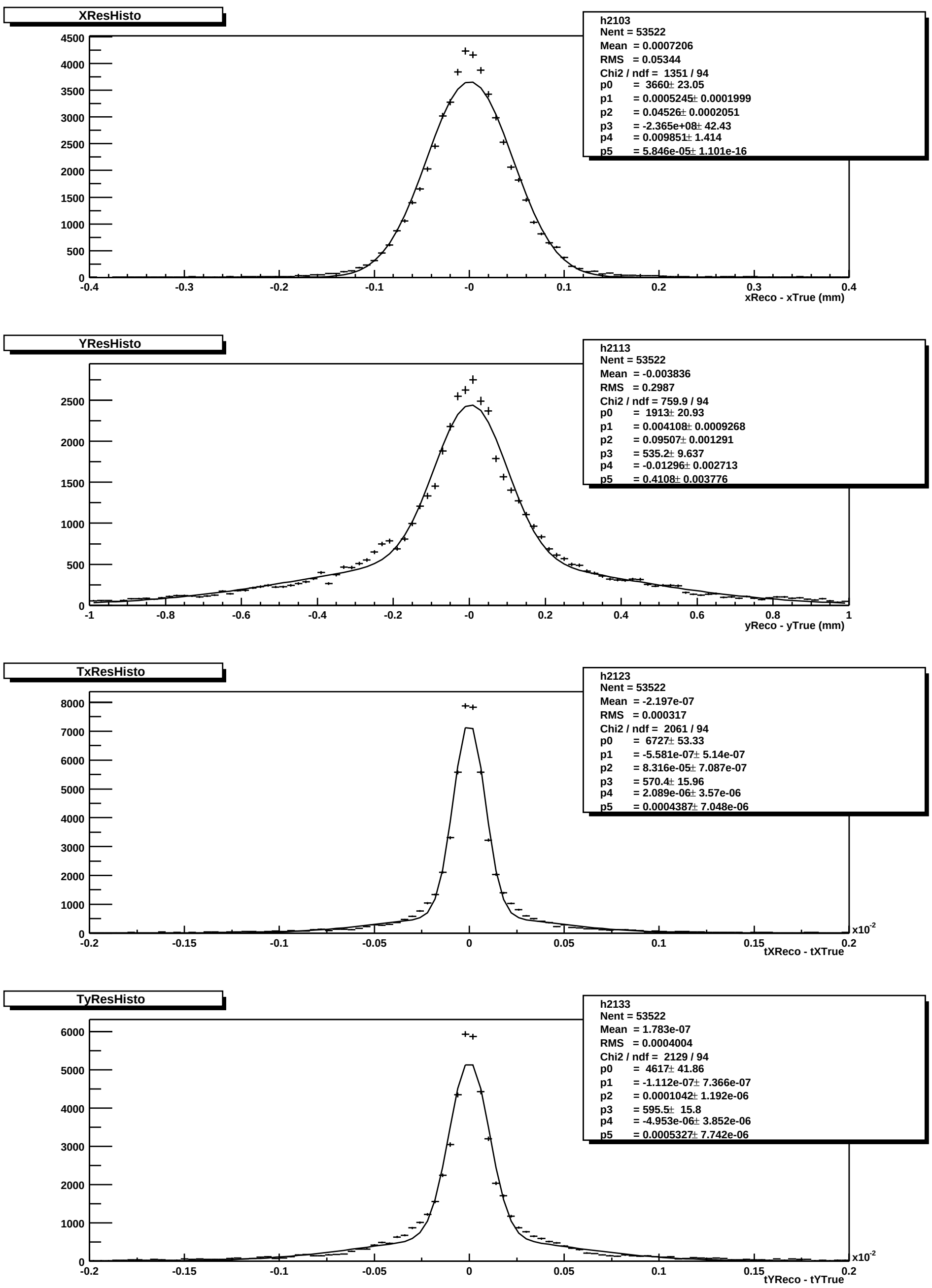

Figure 8: Resolution distribution of the track parameters x, y, t_x, and t_y at the Inner Tracker hit positions. The fitted curve is a double Gaussian.

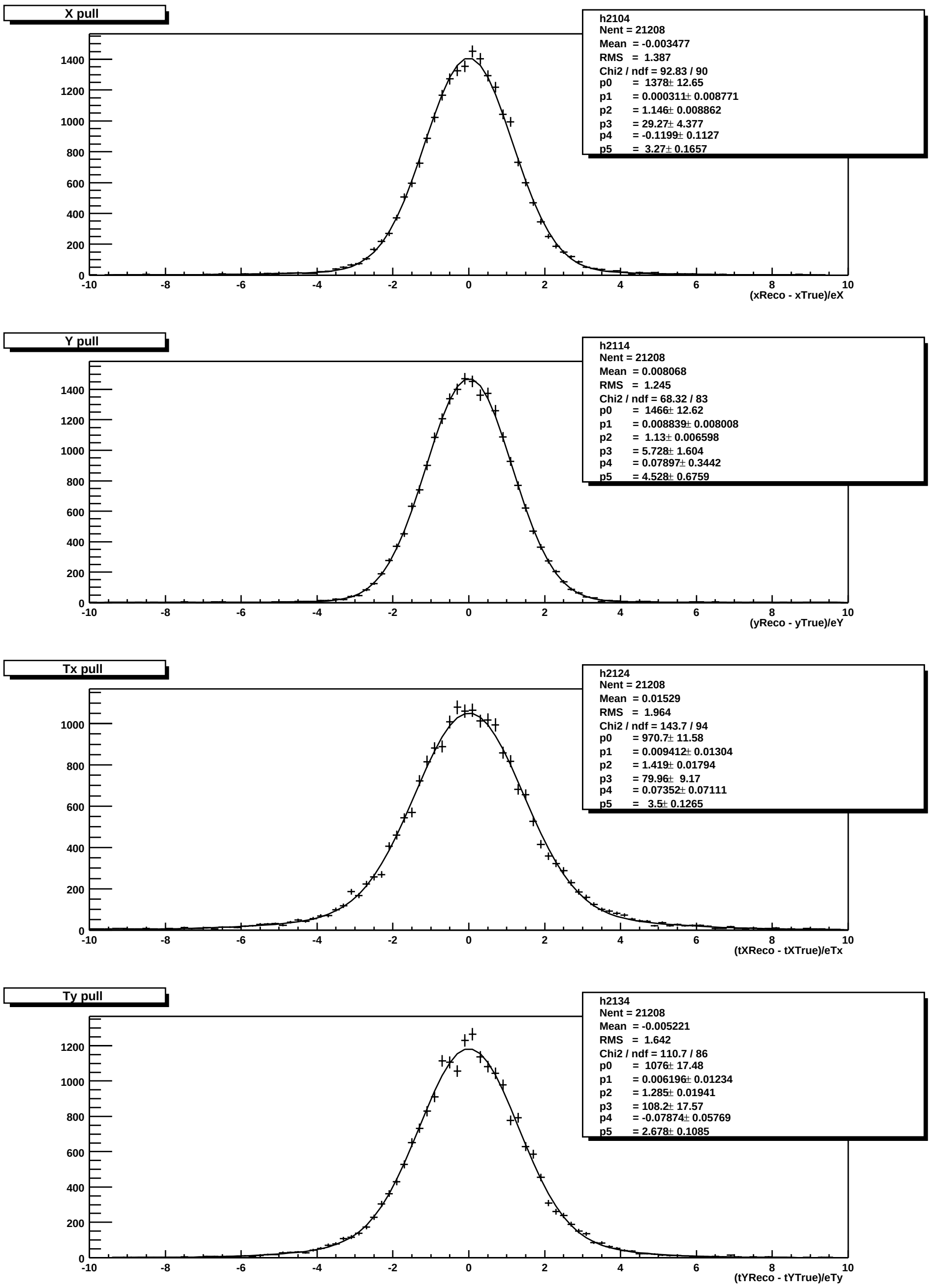

Figure 9: Pull distribution of the track parameters x, y, t_x, and t_y at the vertex detector R-cluster positions. The fitted curve is a double Gaussian.

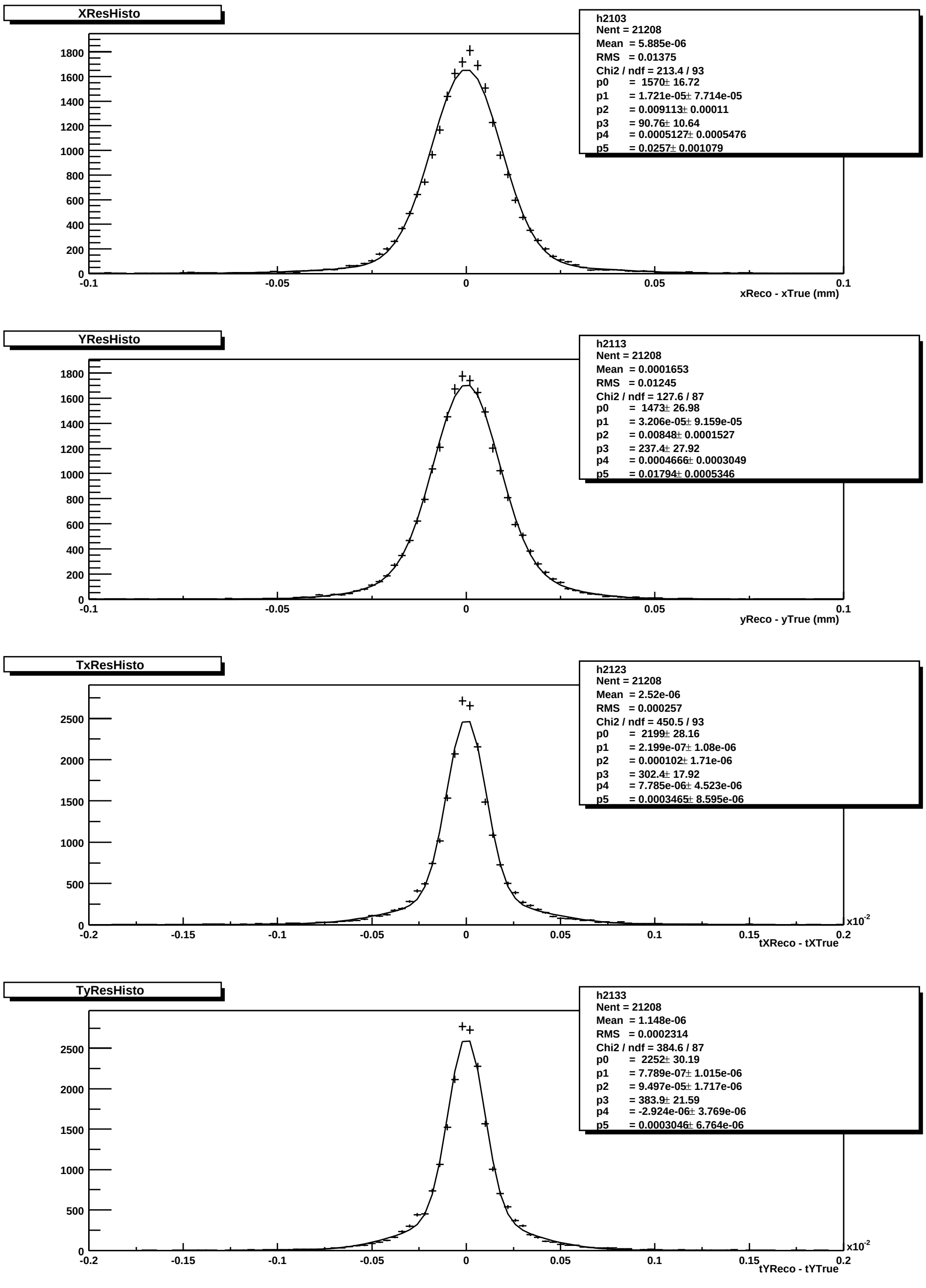

Figure 10: Resolution distribution of the track parameters x, y, t_x, and t_y at the vertex detector R-cluster positions. The fitted curve is a double Gaussian.

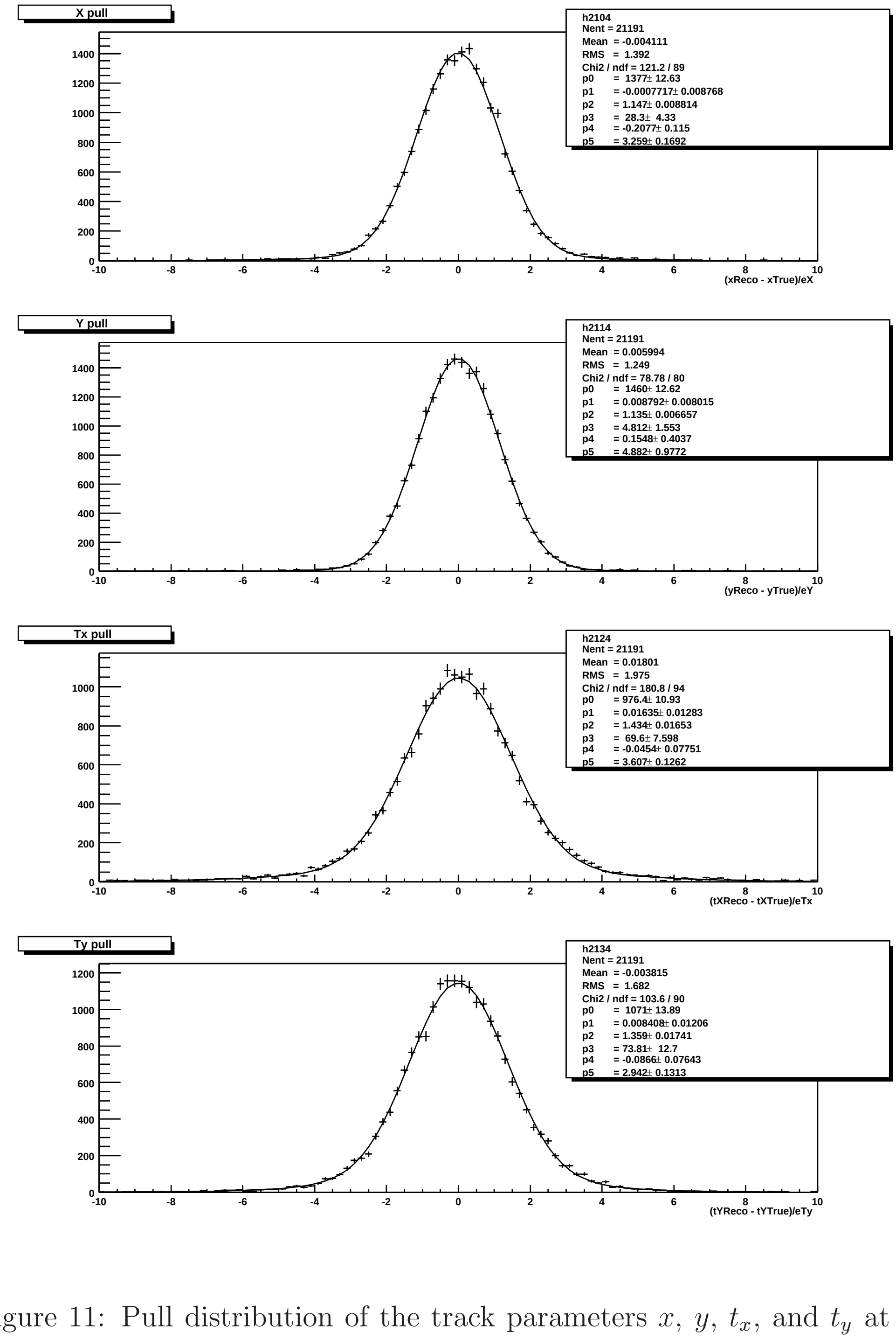

Figure 11: Pull distribution of the track parameters x, y, t_x, and t_y at the vertex detector ϕ-cluster positions. The fitted curve is a double Gaussian.

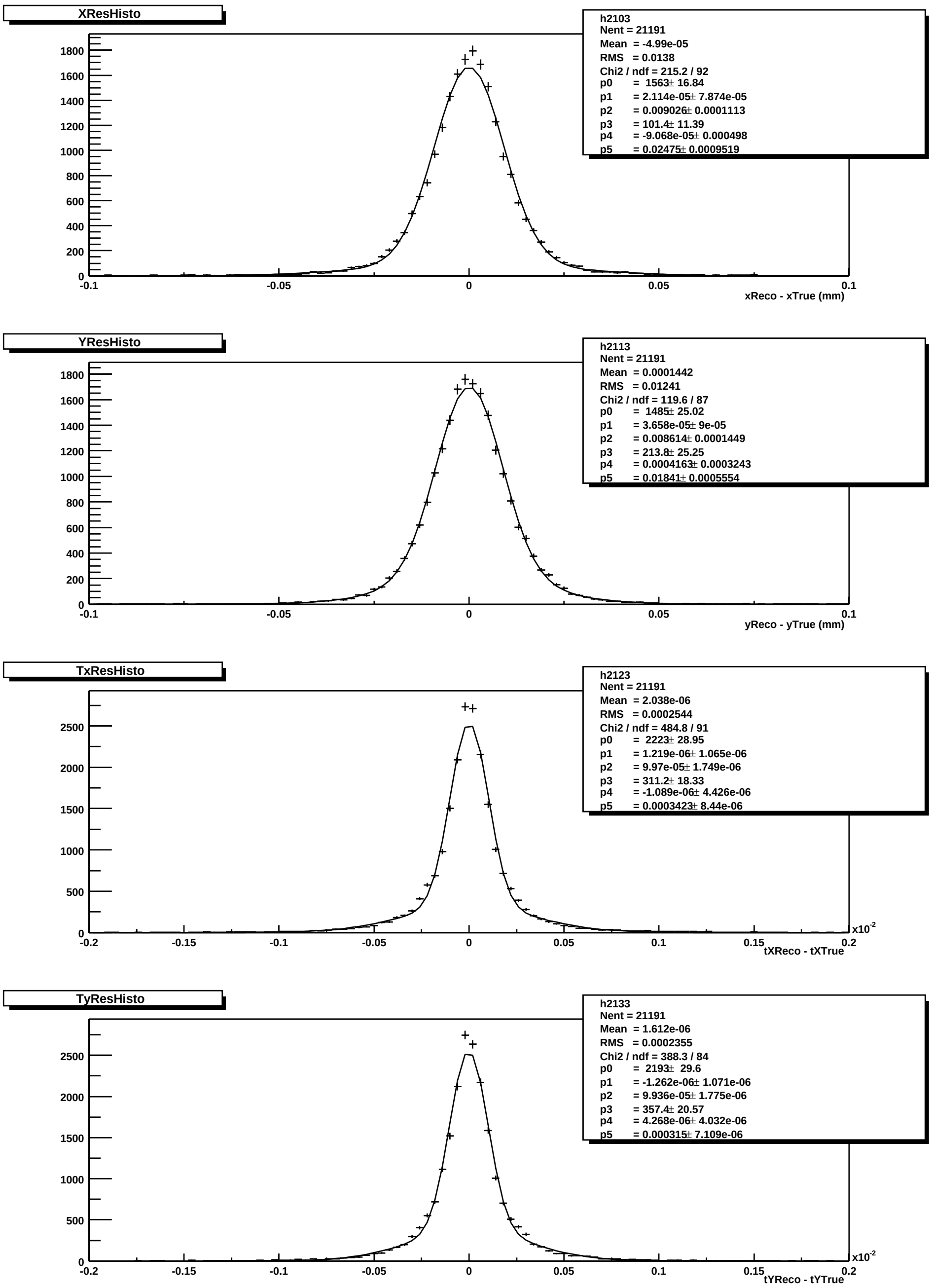

Figure 12: Resolution distribution of the track parameters x, y, t_x, and t_y at the vertex detector ϕ-cluster positions. The fitted curve is a double Gaussian.

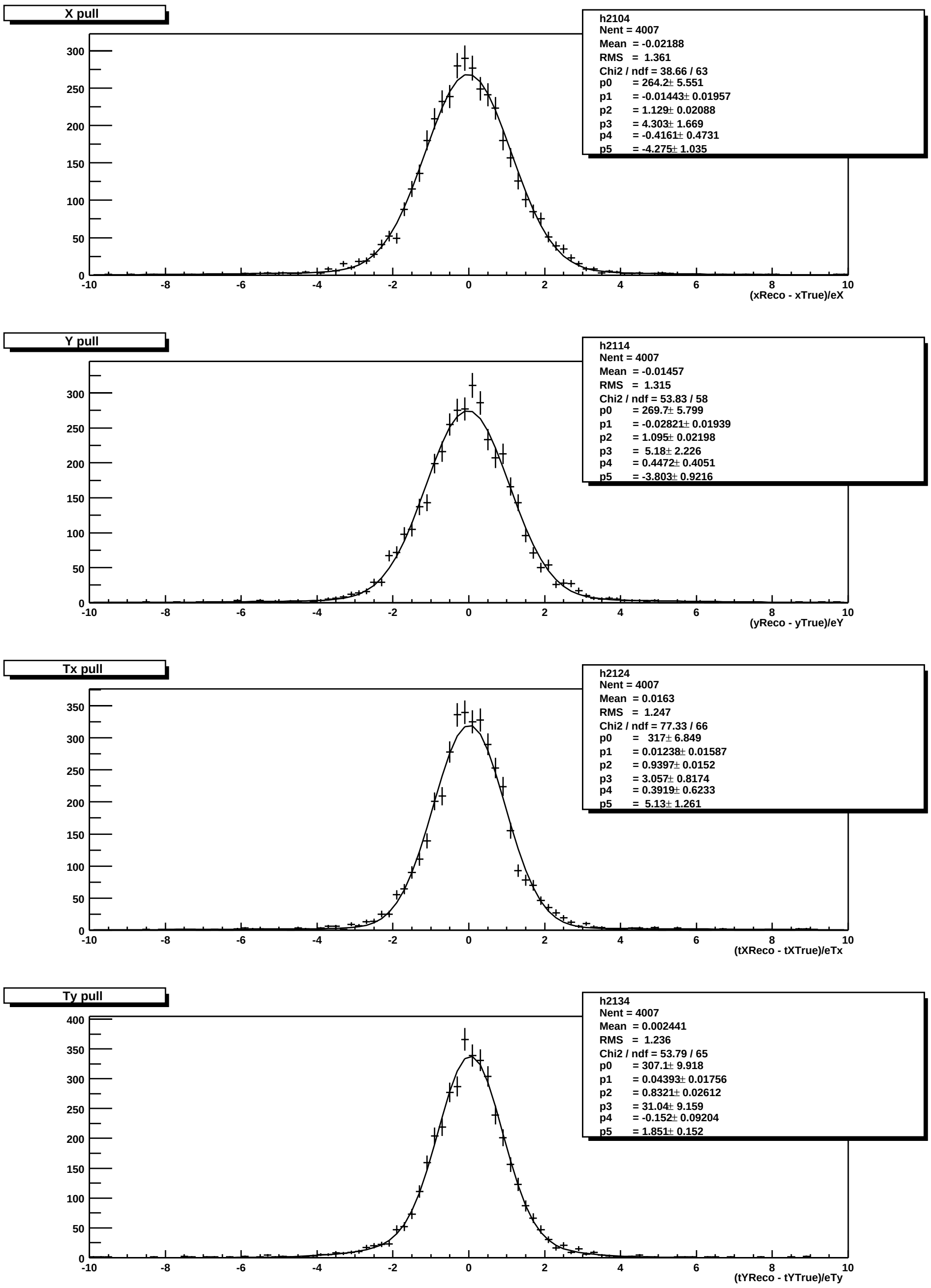

Figure 13: Pull distribution of the track parameters x, y, t_x, and t_y at the true track creation vertex position. The fitted curve is a double Gaussian.

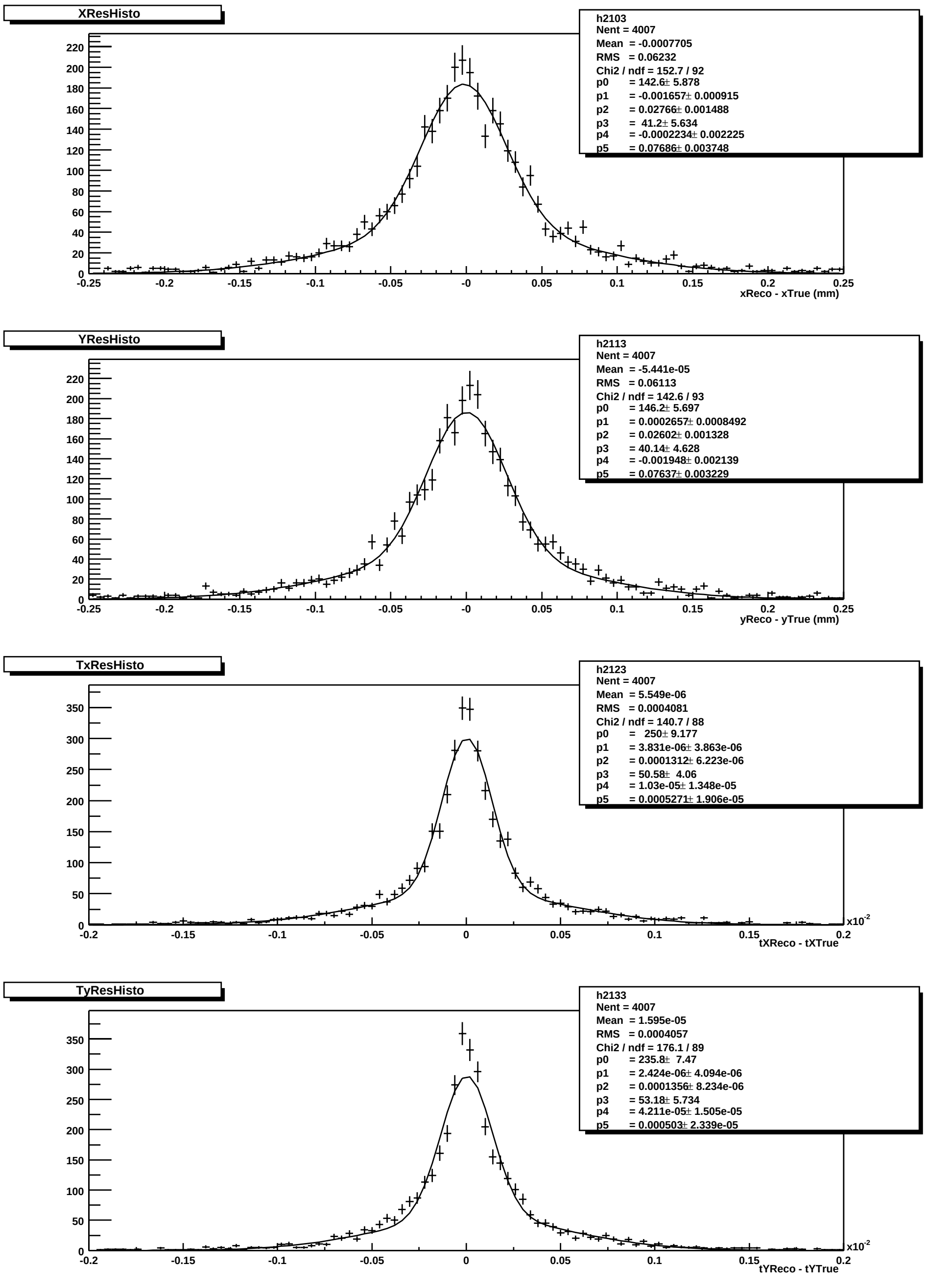

Figure 14: Resolution distribution of the track parameters x, y, t_x, and t_y at the true track creation vertex position. The fitted curve is a double Gaussian.

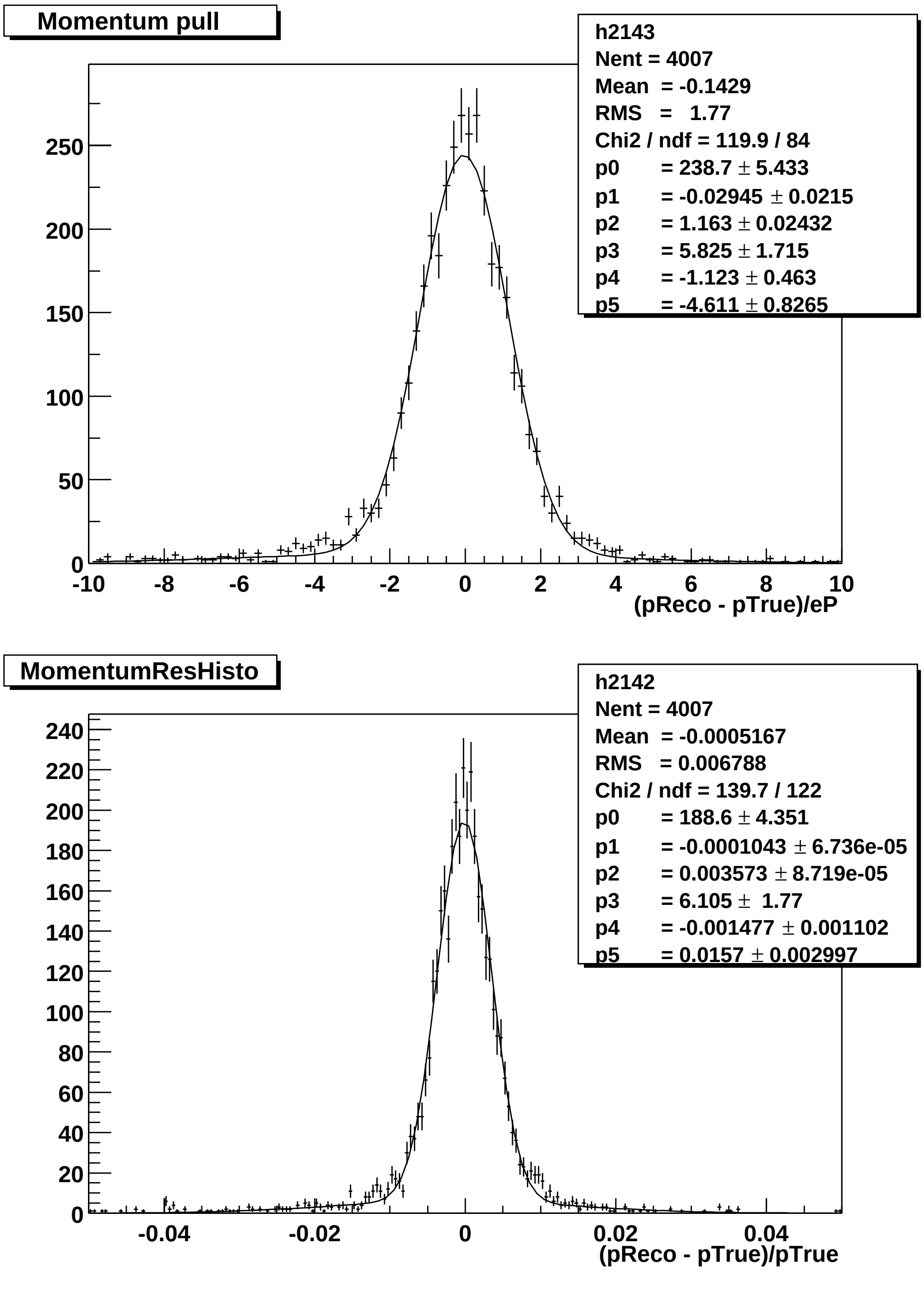

Figure 15: Resolution and pull distribution of the track momentum at the true track creation vertex position. The fitted curve is a double Gaussian.

References

[1] R. van der Eijk et. al., "Performance of the LHCb OO Track Fitting Software", LHCb 2000-086, August 2000

[2] N. van Bakel et. al., "VELO Geometry Optimisation", LHCb 2000-090, October 2000.

[3] D. Steele, "The R-ϕ Silicon Vertex detector Simulation as Implemented in SICB"

[4] M. Frank and B. Hay, "Accessing Level 1 and Velo SICb data within the GAUDI Framework, Classes and access methods", LHCb 2000-009, February 2000.

[5] R. Mankel, "Application of the Kalman Filter Technique in the HERA-B Track Reconstruction", HERA-B 95-239, 1995.

[6] A. Tsaregorodtsev, "SICB User Guide", 1997.

[7] M. Cattaneo, "Brunel User Guide", 2001.